REMARKS

Claims 1-29 are still pending in this application.

Reconsideration of the application is earnestly requested. The Examiner is thanked for the telephone interview of November 14, 2006. The reasons presented at the interview that warrant favorable action are presented below.

A replacement drawing sheet to correct Figure 13 is enclosed and has the changes incorporated. An annotated sheet showing the changes informally is also included to aid the Examiner. Figure 13 has been amended to include an additional decision step after block 558 and an additional decision step after block 562, as required by the Examiner. These additional decision steps are described in the specification at pages 22-23 and elsewhere; no new matter has been added.

The Examiner has rejected claims 1-18 under §103 as being unpatentable over *Hunt et al.* (*Hunt*) in view of *Masinter*. Although the Examiner's arguments have been carefully considered, Applicant respectfully traverses these rejections as explained below.

Claim 1

Claim 1 requires that two different hash functions are used to calculate two different values for a computer file. (The specification makes clear that the hash functions are different. Further, claim 6 requires that the second hash function provide stronger security; inherently then, the second hash function must be different than the first.) The claim specifically requires:

computing a first hash value for said file using a first hash function;
computing a second hash value for said file using a second hash function.

Hunt does disclose that a cryptographic hash function is used to generate a message digest (or hash value) based upon the contents of a file (paragraph 27). But, *Hunt* only discloses that a single digest is computed for each computer file (see also paragraphs 28, 29 and 30). In addition, while *Hunt* also discloses that a digest might represent more than one file (paragraph 31), there is no discussion whatsoever that a second hash function is used to generate a second hash value for the same computer file as required by claim 1.

The Office action cites paragraph 32 and Figure 6 as disclosing using a second hash function to compute a second hash value for the computer file. But, paragraphs 32 and 33 make

2

clear that this additional message digest generated is not a second digest for the <u>same computer file</u>, it is a message digest for a <u>list of message digests</u> which is an entirely different computer file: "A message digest is then generated for the list of these message digests" (paragraph 33, lines 2, 3), "By generating a message digest for a list of message digest of all files on the client" (lines 11, 12), "where all message digests for the individual files are combined into a single file" (paragraph 34, lines 10, 11), and "a message digest is generated for the file containing the message digests for the individual files" (paragraph 34, lines 15, 16). Claim 1 specifically requires that the second hash value is computed using a second hash function for the <u>same</u> computer file for which a first hash value was computed.

Claim 1 also requires:

adding said first hash value and said second hash value to a data structure associated with said database.

As mentioned directly above, *Hunt* does not disclose the second hash value computed from the same computer file. Further, the claim requires that both values are added to a data structure associated with the database. The Office action cites paragraphs 31 and 32 as disclosing two message digests being stored in the repository database. To the contrary though, although the final sentence of paragraph 31 discloses that the message digest (for a particular file) is copied from the client to the database repository, the message digest of the list of message digests is never copied or stored anywhere, and certainly is not stored anywhere together with the first message digest. Therefore, *Hunt* does not disclose two hash values (or message digests) for a single computer file being added to a data structure that is associated with the database.

It is respectfully submitted that the above steps are not taught or suggested by *Hunt* and it is requested that the rejection of claim 1 be withdrawn.

<u>Claim 7</u>

Claim 7 is a method of retrieving a computer file from a database and requires retrieving a first hash value-second hash value pair from a data structure. The first hash value is derived from the stored file using a first hash function and the second hash value is derived from the stored file using a second hash function. Thus, the first hash value is used as a reference to address and store the file (the unique identifier matches the first hash value) and the second hash value may be used to verify the contents of the file. There are two hash functions and two hash

3

values, respectively, that are associated with the stored file. As explained above, *Hunt* does not disclose two different hash values for a single computer file.

The Office action also relies upon *Masinter*. But, *Masinter* also only discloses a single hash function being used for a particular file. Figure 1 only discloses that a single hash 16 is calculated for a particular document 18 (or for certain attributes of that document). Column 4, lines 6-60 likewise discloses that only a single hash value is calculated for a particular document, and then that hash value is used to find a location for that document. There is no disclosure in *Masinter* of two hash values being calculated for a given file, where the second hash value is used to verify the contents of the file.

For lease these reasons, it is requested that the rejection of claim 7 be withdrawn.

Claim 13

Claim 13 is a method of adding hash authority functionality to a database of files and requires that an addressing hash function be used to compute an addressing hash value for a first file, and requires that a verification hash function is used to compute a verification hash value for said first file as well. Both the addressing hash value and the verification hash value are added to an entry of the data structure and are thus associated with one another in the database.

As explained above, neither *Hunt* nor *Masinter* disclose two different hash functions being used to calculate two different hash values for a particular file in a database. Further, both references do not disclose associating these two hash values with each other and adding them to a structure that is associated with the database.

For at least these reasons, it is requested that the rejection of claim 13 be withdrawn.

Claim 17

Claim 17 is a method of upgrading a verification hash function for a database of files. The claim requires a data structure representing the database; the data structure includes an addressing hash value-verification hash value pair for each of the files in the database. Each addressing hash value is computed using an addressing hash function and each verification hash value is computed using a verification hash function. As explained above, neither of the cited references discloses two different hash functions and two different hash values being computed for a single file. Further, claim 17 requires that a new verification hash value be computed for

4

each file using a stronger verification hash function. There is no disclosure whatsoever in either reference of upgrading to a new verification hash function in this manner. Both references only disclose using a single hash function.

For at least these reasons, it is requested that the rejection of claim 17 be withdrawn.

The Examiner has rejected claims 19-29 under §103 as being unpatentable over *Hunt et al.* (*Hunt*) in view of *Castro*. Although the Examiner's arguments have been carefully considered, Applicant respectfully traverses these rejections as explained below.

Claim 19

Claim 19 requires generating a random number for a computer file and then storing the file in a database at a location identified by the random number. In other words, the random number is used for addressing purposes. The Office action relies upon *Castro* as teaching storing a file using a random number as an address and relies upon paragraph 36. To the contrary, this paragraph only explains that files may be stored on a variety of storage devices and there are numerous factors used to determine which device will store which files. Those factors include the number of devices, the storage space for each device, and a <u>single</u> random number. Thus, a single random number is used along with numerous other factors to determine which files will be stored on which particular device. A random number is <u>not</u> generated for each file to be stored. Further, the single random number used is not being used as an address to identify a location for storing a file as specifically required by claim 19.

Claim 19 also requires that the random number for the computer file and the verification hash value for the computer file are added together as an entry in a data structure and are associated with one another. Neither of the cited references disclose a random number and a hash value for a particular file being generated and stored together in this manner.

For at least these reasons, it is requested that the rejection of claim 19 be withdrawn.

Claim 22

Claim 22 requires a data structure with a random number-verification hash value pair for each file in a database. As explained above, neither of the cited references disclose a random number being generated for file, a verification hash value being generated for that file, and then

5

both of these values being associated together in a data structure as representing a file of the database. For at least this reason, is requested that the rejection of claim 22 be withdrawn.

Claim 24

Claim 24 is a method of adding a computer file to a database, a first random number being generated for the computer file. As explained above, neither of the cited references disclose a random number being generated for a file, a verification hash value being generated for that file, and then both of these values being associated together in a data structure as representing a file of the database. Further, the claim requires adding a mapping to the entry that maps the first random number to the second random number. As shown in Figure 12, such a technique is useful when a duplicate copy of the file is added to a database. For at least these reasons, is requested that the rejection of claim 24 be withdrawn.

Claim 26

Claim 26 is a method of retrieving a file from a database, where the stored file is retrieved from the database using a random number as a reference. The claim also requires that a hash value for the file is associated with the random number in the data structure. As explained above, none of the cited references disclose using a random number as a reference address for a file, nor associating the random number with a hash value in a data structure as representing a file in a database. For at least these reasons, is requested that the rejection of claim 26 be withdrawn.
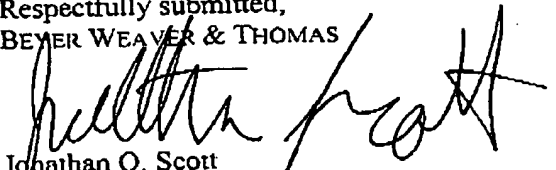
Dependent Claims

Since the dependent claims depend on the independent claims, it is respectfully submitted that they are each patentable over the art of record for at least the same reasons as set forth above. Further, each of the dependent claims require additional features that when considered in light of the claimed combination further distinguish the claimed invention from the art of record. For example, claim 2 specifically requires computing a verification hash value for a copy of the file using the second hash function. Neither reference discloses a second hash function being used in this way.

Claims 6, 12 and 15 require that the second hash function provide stronger security than the first hash function. None of the cited references disclose a second hash function being

6

computed for a particular file that is a stronger hash function than the first hash function for a particular file.

Reconsideration of this application and issuance of a Notice of Allowance at an early date are respectfully requested. If the Examiner believes a telephone conference would in any way expedite prosecution, please do not hesitate to telephone the undersigned at (612) 252-3330.

Respectfully submitted,
BEYER WEAVER & THOMAS

Jonathan O. Scott
Registration No. 39,364

BEYER WEAVER & THOMAS, LLP
P.O. Box 778
Berkeley, CA 94704-0778

Telephone: (612) 252-3330
Facsimile: (612) 825-6304

7